Copilot

Certainly! Below is an example of how you can read data from an XML file and display it in a tabular format using JavaScript. We'll assume you have an XML file named `employee.xml` with employee details. Let's create an HTML file called `index.html` to achieve this:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
        /* Style the table */
        table {
            border-collapse: collapse;
            width: 100%;
        }
        th, td {
            border: 1px solid #ddd;
            padding: 8px;
            text-align: left;
        }
        th {
            background-color: #f2f2f2;
        }
    </style>
</head>
<body>
    <h1>Employee Details</h1>
    <table>
        <thead>
            <tr>
                <th>ID</th>
                <th>First Name</th>
                <th>Last Name</th>
                <th>Title</th>
                <th>Division</th>
                <th>Building</th>
                <th>Room</th>
            </tr>
        </thead>
        <tbody id="employeeTableBody"></tbody>
    </table>

    <script>
        // Read and parse the XML file
        const xhr = new XMLHttpRequest();
        xhr.open("GET", "employee.xml", true);
        xhr.onreadystatechange = function() {
            if (xhr.readyState === 4 && xhr.status === 200) {
                const xmlDoc = xhr.responseXML;
                const employees = xmlDoc.getElementsByTagName("employee");

                // Populate the table
                const tableBody = document.getElementById("employeeTableBody");
                for (const emp of employees) {
                    const id = emp.getAttribute("id");
                    const firstName = emp.getElementsByTagName("firstname")[0].textContent;
                    const lastName = emp.getElementsByTagName("lastname")[0].textContent;
                    const title = emp.getElementsByTagName("title")[0].textContent;
                    const division = emp.getElementsByTagName("division")[0].textContent;
                    const building = emp.getElementsByTagName("building")[0].textContent;
                    const room = emp.getElementsByTagName("room")[0].textContent;

                    const row = document.createElement("tr");
                    row.innerHTML = `
                        <td>${id}</td>
                        <td>${firstName}</td>
                        <td>${lastName}</td>
                        <td>${title}</td>
                        <td>${division}</td>
                        <td>${building}</td>
                        <td>${room}</td>
                    `;
                    tableBody.appendChild(row);
                }
            }
        };
        xhr.send();
    </script>
</body>
</html>
```

Explanation:

- We create an HTML table to display the employee details.
- The JavaScript code fetches the `employee.xml` file using an XMLHttpRequest.
- It then parses the XML data and dynamically populates the table rows with employee information.

Make sure to adjust the XML file content and styling according to your needs! 😊

For more details on parsing XML in JavaScript, you can refer to this GeeksforGeeks article.